Lab on a Chip

Cite this: *Lab Chip*, 2012, **12**, 1841

www.rsc.org/loc

**PAPER**

# Biomolecular theorem proving on a chip: a novel microfluidic solution to a classical logic problem†

**Seung Hwan Lee,‡[a] Danny van Noort,‡[b] Kyung-Ae Yang,[c] In-Hee Lee,[c] Byoung-Tak Zhang*[c] and Tai Hyun Park*[a]**

***Received 23rd July 2011, Accepted 24th February 2012***
**DOI: 10.1039/c2lc20677c**



Biomolecules inside a microfluidic system can be used to solve computational problems, such as theorem proving, which is an important class of logical reasoning problems. In this article, the Boolean variables (literals) were represented using single-stranded DNA molecules, and theorem proving was performed by the hybridization and ligation of these variables into a double-stranded "solution" DNA. Then, a novel sequential reaction mixing method in a microfluidic chip was designed to solve a theorem proving problem, where a reaction loop and three additional chambers were integrated and controlled by pneumatic valves. DNA hybridization, ligation, toehold-mediated DNA strand displacement, exonuclease I digestion, and fluorescence detection of the double-stranded DNA were sequentially performed using this platform. Depending on the computational result, detection of the correct answer was demonstrated based on the presence of a fluorescence signal. This result is the first demonstration that microfluidics can be used to facilitate DNA-based logical inference.

## Introduction

Biomolecular computing involves a multidisciplinary approach consisting of molecular biology, microsystems, biosensing, and information science. This type of computing uses biological molecules, usually DNA, to perform selections by simulating the digital information processing procedures.[1–3] Since biological molecules are based on solution reactions, biomolecular computing is best performed on a fluidic platform.

Microfluidic technologies have emerged as promising devices for chemical reactions and biological analysis. In the field of biomolecular computing, microstructures have been used to control the flow of solutions, and as such, the flow of information.[4–6] The advantages of this technology include low sample and reagent costs, high reaction efficiency, fast reaction rate, high throughput, simple operation and precise control.[7] Furthermore, the structures can be designed to be problem-specific or reconfigurable.[8,9] By using valves and pumps in the fluidic network, the computer architecture can be problem-independent and autonomous.[10] Because of these advantages, microfluidics has been used as hardware for biomolecular computing, especially DNA computing. Several groups have proposed using microfluidic platforms to solve maximal clique problems,[9] logic operations,[11] satisfiability problems,[5] and splicing model based problems.[12]

In these DNA computing chips, DNA hybridization analysis is very important for obtaining computation results. Most DNA computing microfluidic chips use an intercalating dye as the detection tool.[5,10] However, since intercalating fluorescence dyes label all double-stranded DNAs "solutions", including nonspecific products, other microfluidic platforms have been developed to employ fluorescence resonance energy transfer (FRET) as a sequence-specific detection tool, including a labeled probe and labeled target oligomer,[13] molecular beacon,[14,15] and two fluorescently labeled probes.[16] FRET is based on the nonradiative transfer of excitation from a donor fluorophore to an acceptor. The fluorescence signal differs according to the distance between the donor and the acceptor. Recently, an alternative double-stranded DNA probe was developed as a detection method using FRET.[17] As the FRET of the double-stranded DNA probe is based on the thermodynamics of DNA, the probe shows a different fluorescence signal depending on the target DNA. Therefore, this probe system can be used as a new sequence-specific detection tool of DNA computing.

In this study, a classical artificial intelligence (AI) problem was solved using a microfluidic chip. Several groups have proposed biomolecular computing solutions to logical reasoning.[18–22] Theorem proving, for example, is a method of logical inference in

[a]*School of Chemical and Biological Engineering, Bio-MAX Institute, Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul 151-744, Korea. E-mail: thpark@snu.ac.kr; Fax: +82-2-875-9348; Tel: +82-2-880-8020*
[b]*MechanoBiology Institute, T-Lab, National University of Singapore, 5A Engineering drive 1, Singapore 117411, Singapore*
[c]*School of Computer Science and Engineering, Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul 151-744, Korea. E-mail: btzhang@snu.ac.kr*

† Electronic supplementary information (ESI) available. See DOI: 10.1039/c2lc20677c

‡ The first and the second authors equally contributed to this work.

propositional logic. It has a variety of applications, including diagnosis and decision making.[23,24] Resolution refutation has been used to prove the theorem, which is a general technique that starts with a set of logical formulae containing a negated conclusion to derive an inconsistency of the formula set. To solve this problem, a new sequential reaction chip was designed using PDMS pneumatic valves to control the flow of information. Thus, refutation was performed in an automated microfluidic system, in which the variables, which were represented by DNA sequences, were mixed with ligation enzymes. To detect the computational results, toehold-mediated DNA strand displacement and enzyme digestion were used.

Moreover, a FRET-based double-stranded DNA probe was used as a tool to measure the DNA computing results on a single chip.

## Theory: resolution theorem proving

Theorem proving is an automated reasoning from the propositional logic.[23,24] Resolution refutation is a general technique to prove a theorem. The goal of theorem proving by resolution refutation is to determine the consistency of a given set of logic formulae using resolution deduction.[24]

Propositional logic formula consists of Boolean variables and logic connectives. A Boolean variable is called a literal, while the basic logic connectives are $\wedge$ (logical AND), $\vee$ (logical OR) and $\neg$ (logical NOT). To solve this problem, every formula should be expressed in the so-called "clause" form. A clause is defined as a set of literals connected by $\vee$. A clause form in propositional logic is defined as clauses connected only by $\wedge$. The resolution operates with two clauses sharing one literal and its negation (logical NOT, $\neg$), such as $A \vee B$ and $\neg A \vee C$. From these two clauses, a new clause that is a union of the two clauses can be deduced, except for the shared literal.[25] Thus, a new clause, $B \vee C$, can be made, except for the deduced A, $\neg A$ from $A \vee B$ and $\neg A \vee C$. This outcome of the resolution is called a resolvent (Fig. 1a).

The basic idea of refutation is to show that the negation of the conclusion is inconsistent with the premises. If the result is inconsistent, an empty clause would remain. A simple resolution theorem proving problem was developed. Consider a set of formulae $\Delta = [\neg A_0, A_0 \vee B_0, A_1 \vee B_1, A_i \vee B_i, A_n \vee B_n, \neg B_0 \vee \neg A_1, \neg B_{i-1} \vee \neg A_i, \neg B_{n-1} \vee \neg A_n, \neg B_n \vee \neg C, C]$ ($i = 1, 2, \ldots, n$). In this study, we dealt with an example when $n = 1$. In these formulae, the "$\neg A_0$" means the negation of the conclusion. To prove "$A_0$" is consistent with $\Delta$, the "$\neg A_0$" was added to the formulae.

As shown in Fig. 1b and c, the clauses and their resolvents are connected with lines. In Fig. 1b, the resolution between $A_0 \vee B_0$ and $\neg A_0$ results in $B_0$. $\neg B_0 \vee B_1$ is resolved from $A_1 \vee B_1$ and $\neg B_0 \vee \neg A_1$. From $\neg B_1 \vee \neg C$ and C, a new clause, $\neg B_1$, is also made. By using these resolvents ($B_0$, $\neg B_0 \vee B_1$ and $\neg B_1$), the resolution process is continued again. From $\neg B_0 \vee B_1$ and $\neg B_1$, $\neg B_0$ can be deduced. Finally, an empty clause is produced from $B_0$ and $\neg B_0$. This resolution process can be continued until an empty clause is produced, denoted by the symbol nil. This means that a contradiction occurs for the negation of the conclusion, $\neg A_0$. Thus, it is proved that $\Delta$ is inconsistent. If any clause is not involved in the resolution process, no new resolvent can be produced. For example, if $A_1 \vee B_1$ is not involved in the resolution process, the final resolvent $\neg A_1$ and $\neg B_1$ are made as shown in Fig. 1c. Then, $\neg A_1$ and $\neg B_1$ do not have any shared literal. Thus, the resolution process stops. This case cannot be driven to nil and then it is proved that $\Delta$ is consistent.

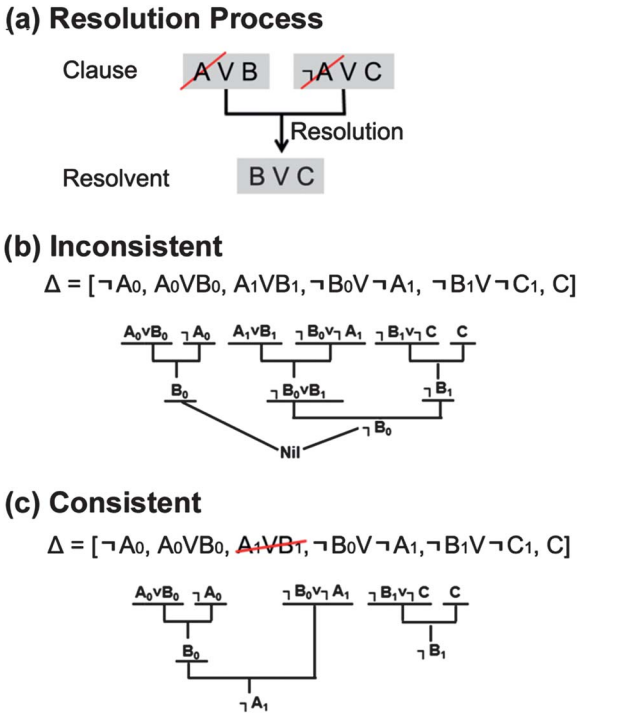


**Fig. 1** Resolution theorem proving. The basic logic connectives used here are $\wedge$ (logical AND), $\vee$ (logical OR) and $\neg$ (logical NOT). (a) Resolution process. The clauses and their resolvents are connected with lines. The resolution operates with two clauses sharing one literal and its negation. $B \vee C$ is derived from $A \vee B$ and $\neg A \vee C$. The shared literal A and its negation $\neg A$ are removed and new clause $B \vee C$ is made. (b) A given set of formulae for inconsistent case and its resolution theorem proving process for the set of clauses. The clauses and their resolvents are connected with lines. All clauses participate in this process. An empty clause (nil) is derived by the resolution operation. This proves that $\Delta$ is inconsistent. (c) A given set of formulae for consistent case and its resolution theorem proving process. $A_1 \vee B_1$ is not involved in the resolution process. Therefore, the resolution process stops before reaching the empty clause (nil). This means that $\Delta$ is consistent.

In this paper, resolution theorem proving was performed for the proposed simple formulae, $\Delta = [\neg A_0, A_0 \vee B_0, A_1 \vee B_1, \neg B_0 \vee \neg A_1, \neg B_1 \vee \neg C, C]$. Based on the formulae, the consistent and inconsistent cases were prepared. The inconsistent and consistent cases were same as shown in Fig. 1b and c. However, for the consistent case, each clause was excluded one by one from the resolution process.

## DNA computing process for resolution theorem proving

Depending on the class of the problem, resolution theorem proving may require an exponential run time.[26] Lee *et al.*[27] developed a method using representation in a DNA molecule to effectively solve the theorem proving problem. According to this

DNA computing process, the literals were represented by short single-stranded DNA sequences. The negations of the literals were represented by the complementary sequence of the positive literals. The clauses were represented as the concatenation of sequences representing the variables and negations. An illustrative example is shown in Fig. 2a.

The literals for the logic problem were encoded by 10 different oligonucleotides of 10 nucleotides (nt) each (Table S1, ESI†), while the four clauses were represented by 20 nt sequences combining two literals and two other clauses representing the literals (Table S2, ESI†). The DNA sequences were designed using the sequence design program NACST (Nucleic Acid Computing Simulation Toolkit),[28] considering the minimal tendency of cross-hybridization and the maximal differences between them. The literal "$\neg A_0$" (NOT $A_0$) oligonucleotide contained an additional 10 nt sequence as a negation of the conclusion and initiation point of the detection step.

The resolution of a variable between two clauses is represented by the hybridization of the two short DNA sequences, which corresponds to the literal in each clause. For example, when a variable Y is resolved from clauses A and B, the DNA sequences and their complementary sequences corresponding to Y or $\neg$Y in each clause hybridize. Therefore, if the result is an empty clause, none of the DNA sequences representing the literals would remain single-stranded (Fig. 2a).

The ligation method was used as an AND operation between clauses to solve a propositional logic problem with DNA. As shown in Fig. 1b, Δ was inconsistent and an empty clause (nil) was reached *via* resolution.[25] In this case, a complete double-stranded structure would be built by the DNA hybridization and ligation (Fig. 2a-i). The formation of a complete double-stranded structure means inconsistency. However, if Δ was consistent an incomplete structure (a partially hybridized and ligated DNA) would be built (Fig. 2a-ii).

## Sequential reaction for DNA-based resolution theorem proving

A sequential reaction was developed to detect the computational results, as shown in Fig. 2b and c. If all the clauses exist, the

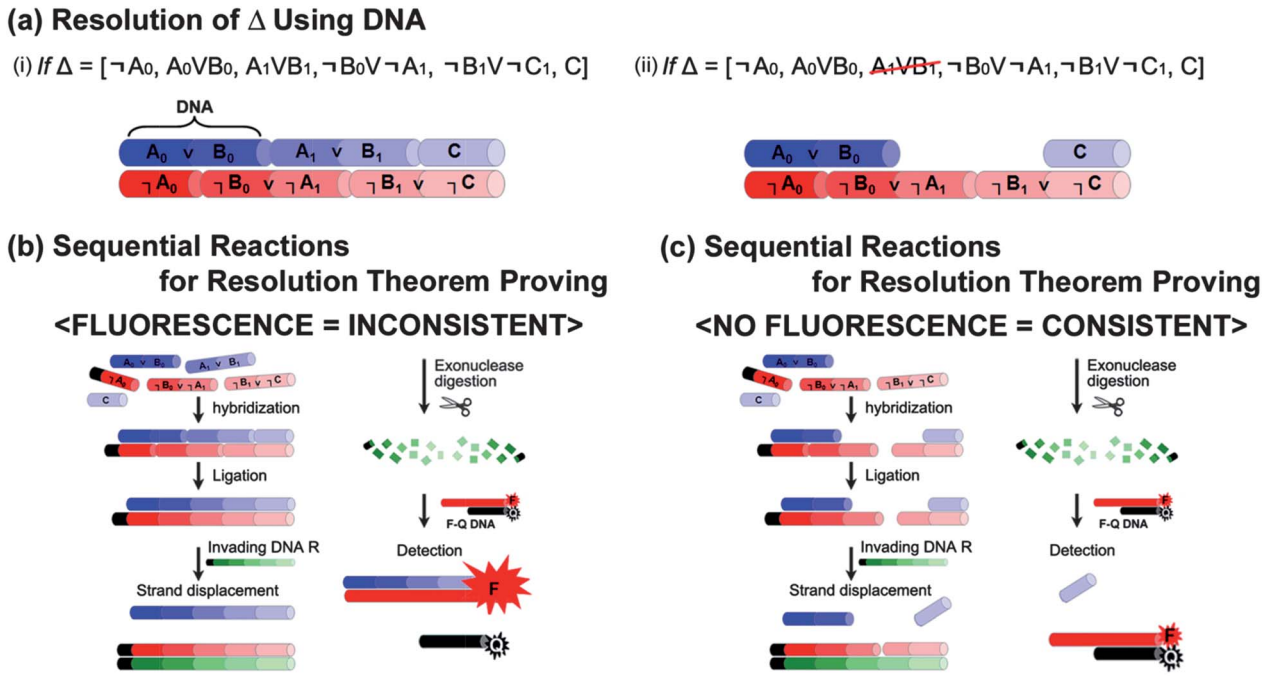


**Fig. 2** Schematics of DNA computing for resolution theorem proving. (a) Resolution theorem proving process for simple example. Each different literal was encoded as a DNA sequence. Negation of literals was also encoded as a complementary sequence of literal DNA. The clauses are shown as the concatenation of sequences representing the variables. When the given set of clauses was logically inconsistent, an empty clause (nil) can be reached by resolution, while the DNA clauses formed a double-stranded DNA structure. The "$\neg A_0$" sequence was the negation of the conclusion containing additional 10 nt sequences for the next steps. With the consistent result, however, the resolution process could not draw an empty clause. Thus, a partially hybridized DNA would be created. (b) Overview of DNA computing for inconsistent case of resolution theorem proving. DNA clauses were hybridized and ligated. If the clauses were logically inconsistent, a double-stranded DNA was formed. The single-stranded DNA R was hybridized with the red-colored sequence. The DNA strand was displaced by branch migration. Exonuclease I degraded the single-stranded DNA from 3′ to 5′. Only the unreacted DNA was digested. The clause "C" sequence was modified by phosphorothioate. Thus, the "$A_0 \vee B_0$, $A_1 \vee B_1$, C" sequence was not removed. The computation result was detected by the double-stranded DNA probe. If the result was inconsistent, the quencher probe of the double-stranded DNA probe was changed by the "$A_0 \vee B_0$, $A_1 \vee B_1$, C" sequence. As a result, a strong fluorescence signal could be detected. (c) Overview of DNA computing for consistent case of resolution theorem proving. A DNA sequence ($A_1 \vee B_1$) was not involved in the process. Incomplete structure (a partially hybridized and ligated DNA) was built. After sequential reactions, the "$A_0 \vee B_0$" sequence was digested by exonuclease I and the "C" sequence remained. The quencher probe of the double-stranded DNA probe was not changed by the "C" sequence. As a result, fluorescence intensity could not be observed.

resolution process of the clauses would draw an empty clause (nil). In other words, if the given set of clauses is logically inconsistent, a hybridized and ligated double-stranded DNA structure would be created. However, if any of the clauses is missing, the resolution process would not be able to draw an empty clause. Thus, a partially hybridized and ligated DNA would be created. To separate these computing results without any intervention, a novel sequential reaction was developed, which detects logical consistency using a fluorescence signal. In other words, if the result is logically inconsistent, a fluorescence signal would be observed. However, if the result is logically consistent, fluorescence intensity would not be observed after the sequential reactions.

For this purpose, the sequential reaction was performed in three steps: (1) toehold-mediated DNA strand displacement; (2) exonuclease I digestion, removing unreacted DNA and (3) detection using the double-stranded DNA probe (Fig. 2b and c). The first step was the DNA strand displacement *via* toehold-mediated reaction. This DNA displacement reaction has been used to construct a variety of DNA devices,[29] including logic gates,[30,31] catalysts,[32,33] and motors.[34] The single-stranded DNA R (invading DNA with a toehold sequence) was prepared (Table S3, ESI†) and mixed with the products of hybridization and ligation. A short single-stranded overhang region (known as a toehold) of the double-stranded DNA (complex DNA) initiated the strand displacement reaction, after which branch migration occurred. After the completion of branch migration, the DNA hybridized with the $\neg A_0$, $\neg B_0 \vee \neg A_1$, and $\neg B_1 \vee \neg C$ sequences. Thus, a single strand of $A_0 \vee B_0$, $A_1 \vee B_1$, C was obtained or not obtained depending on the logical consistency (Fig. 2).

Next, exonuclease I was added to remove unreacted DNAs. Exonuclease I degraded the single-stranded DNA from the 3′ to 5′. To prevent degradation of the computation results, the clause C sequence was modified with a phosphorothioate (Fig. 2b and c).

Finally, a double-stranded DNA probe was prepared and added. The double-stranded DNA probe had a fluorophore-conjugated sequence and a quencher-conjugated sequence. A fluorophore-conjugated probe sequence complementary to the target sequence was prepared. In addition, a quencher-conjugated probe sequence complementary to the fluorophore probe was designed. The fluorophore probe was labeled "Cy3" on the 5′ end while the quencher probe was labeled "blackhole quencher 2" on the 3′ end (Table S3, ESI†). In the absence of a target, the fluorophore and quencher probes hybridized and the distance between the fluorophore and the quencher was reduced, thus, the fluorescence signal was very weak (Fig. 2c). However, the existence of a target would thermodynamically cause the interaction between the quencher and the target molecule to change.[13,15] As a result, a strong fluorescence signal could be detected. In other words, if the computation result was inconsistent, a strong fluorescence signal would be obtained (Fig. 2b).

## DNA computing chip design and fabrication

On a single microfluidic chip, a classical logic problem of resolution theorem proving was solved by DNA computing and the results of the computation were shown based on fluorescence signal intensity. For this purpose, a new chip was designed using PDMS pneumatic valves to control the flow. Fig. 3a shows an overview of the microfluidic device that was used to perform DNA computing. The device was made of two layers of polydimethylsiloxane (PDMS; Sylgard 184, Dow-Corning, MI, USA): the fluidic and pneumatic layers. The bottom layer contained a fluidic channel (100 μm wide and 10 μm deep), while the top layer contained pneumatic channels (100 μm wide, 12.5 μm deep) and valves to control the fluid. The chip was fabricated using standard multilayer soft lithography.[35] This microfluidic chip consisted of a space for four-time sequential reactions. The first reaction occurred in the circulation loop (1.0 mm in diameter), as shown in Fig. 3a. The circular channel was segmented by the pneumatic valves (valves S1 and S2) into two different parts,[36–38] each containing an inlet and an outlet controlled by valves (valves I1, I2, O1, and O2). A peristaltic pump was integrated to mix the different fluids (valves P1, P2, and P3). Next, additional fluid channels, which were segmented by valves (valves S3, S4, S5, and S6), were used as chambers for the second, third, and fourth sequential reactions (Fig. 3b–d). Each chamber contained an inlet and an outlet channel with three twin valves (valves T1, T2, and T3) to control the flow.

## Device operation for DNA-based resolution theorem proving

This chip integrated four main functions and performed them sequentially on a single chip: (1) DNA hybridization and ligation; (2) toehold-mediated DNA strand displacement; (3) exonuclease I digestion, removing unreacted DNA and (4) detection using the double-stranded DNA probe. To perform computation *via* hybridization and ligation, the clauses of the DNA mixture and ligase in a ligation buffer were loaded into each channel of the circular loop, respectively (Fig. 3a). These two solutions were then mixed using the peristaltic pump, without opening chambers 1, 2, and 3.

The sequential reaction occurred after mixing the circular loop solutions. The device contained three additional process chambers. Chamber 1 was used for DNA strand displacement *via* toehold-mediated reaction. Through opening the twin valves on the inlet and outlet channels of the chambers 1 (valves T1), the single-stranded DNA R (invading DNA with a toehold sequence) was injected into chamber 1. After opening the segment valve S3, the DNA R was mixed with the resulting products from hybridization and ligation by using peristaltic valves (Fig. 3b). Next, to remove unreacted DNAs, exonuclease I was injected into chamber 2 by opening the twin valves (valves T2). After closing the twin valves, the segment valves S4 and S5 were opened. Then, the peristaltic mixing was performed (Fig. 3c). Finally, a double-stranded DNA probe was loaded into chamber 3 by opening the valve T3 and mixed by using the segment valve S6 and the peristaltic valves P1, P2, and P3 (Fig. 3d).

## Sequential mixing in a microfluidic chip

Following the development of multilayer soft lithography by the Quake's group, peristaltic mixing using a pneumatic valve has

 

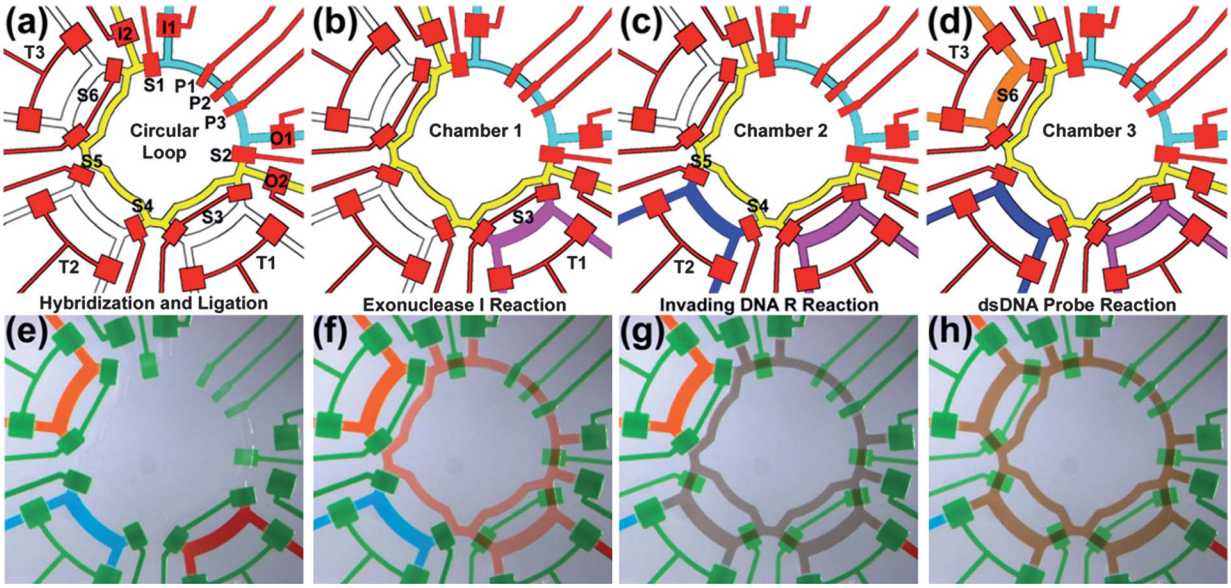


**Fig. 3** (a) Schematic diagram of the sequential reaction chip. The fluid inlet control valves of the circular loop were annotated as I1, inlet 1 and O1, outlet 1. The segmented valves were numbered S1 to S6. The valves for peristaltic mixing were named P1, P2, and P3. The twin valves for controlling the inlet and outlet of the chamber were marked T1, T2, and T3. The chip could be segmented into five sections: DNA clauses and ligase with a buffer were introduced into the circular loop, respectively. In the schematic diagram of the chip, the yellow color represents the ligase solution, and the cyan color represents the clauses of the DNA solution. By mixing, hybridization and ligation occurred. (b) By opening the T1 valve, the DNA strand R was injected (pink color on the chip) into chamber 1. Through opening the S3 valve and operating the peristaltic valves, the DNA R was mixed with the resulting products of the hybridization and ligation. (c) Next, exonuclease I (dark blue) in chamber 2, was mixed by opening the S4 and S5 valves and operating the peristaltic valves. (d) Finally, the double-stranded DNA probe (orange) in chamber 3 was reacted by opening the S6 valve and mixing it using the peristaltic valves. (e) Image of the sequential reaction chip for the mixing of the solution in each chamber. Water was introduced into the circular loop, while red, blue, and orange food dyes were pumped into chambers 1, 2, and 3, respectively. (f) While the peristaltic valves were working, the segment valve for chamber 1 was opened, and red food dye was mixed with water. (g) After the opening of the chamber 2 segmentation valves, the blue food dye was mixed, resulting in a color change. (h) Chamber 3 was opened, and orange food dye was mixed. The contents of the circular loop channel changed color with every step (f, g, and h), indicating successful mixing after each step.



been used as a basic tool for a diverse range of microfluidic chips.[35,39] To verify the mixing between the reagent of the chambers and the solution in the circular loop, food dye was used as an indicator on the chip. After opening the twin valves on the inlet and outlet channels of the chambers (valves T1, T2, and T3), each food dye, which had different colors, was pumped into additional chambers 1, 2, and 3 (Fig. 3e). The twin valves were then closed and the peristaltic pump of the circular loop (valves P1, P2, and P3) was activated. While the solution in the loop was circulating, the segment valve of chamber 1 (valve S3) was opened for the first sequential reaction. After a 1.5 min of mixing, the solutions in the circular loop and chamber 1 were determined to be well mixed, as shown in Fig. 3f. The second sequential reaction occurred when the segment valves (valves S4 and S5) were opened. As shown in Fig. 3g, after 1.5 min, the circular loop and chambers 1 and 2 were filled with the completely mixed solution. Finally, the third sequential reaction was performed as in the previous steps (Fig. 3h). The Movie S1 in the ESI† shows the whole process of the sequential mixing. The designed sequential-reaction chip was demonstrated to be ideally suited for solution-based sequential reactions. Through the sequential reaction on the chip, the computation result was observed using fluorescence signal change of FRET. To the best of our knowledge, there has been no report that utilized FRET, especially for a double-stranded DNA probe, to solve computational problems on microfluidic chips.

## Fluorescence detection of double-stranded DNA probe with different lengths of target DNAs

To evaluate the specificity of the double-stranded DNA probe, five different DNA targets, which had different lengths and were partially complementary to a Cy3 conjugate sequence of double-stranded DNA, were prepared (Table S4, ESI†). This probe is a powerful tool for detecting one base mismatch targets.[17] The computation for solving resolution theorem proving *via* DNA hybridization and ligation resulted in the different lengths of the different target DNAs. As a result of this, a difference of at least 10 nt between inconsistency (perfect-match target) and consistency (partial-match target) was needed to separate the fluorescence signals. As shown in Table S4†, the 50 nt target means inconsistency (nil) and the others (30 and 40 nt) mean consistent DNA computation results. One of the target DNA solutions (25 μM) and the double-stranded DNA probe (Cy3 probe: 25 μM; quencher probe: 50 μM) solution were introduced into the circular loop and the chamber 3 through the inlet channels, respectively. Each solution contained 10 mM $MgCl_2$. After opening the segment valve (valve S6), the mixing of the two solutions occurred by the peristaltic pump. The fluorescence signal was detected using a fluorescent stereo microscope.

Fig. 4 shows a comparison of the FRET emission from the reaction between the double-stranded DNA probe and the target DNAs. When the 50 nt target sequence was used, the

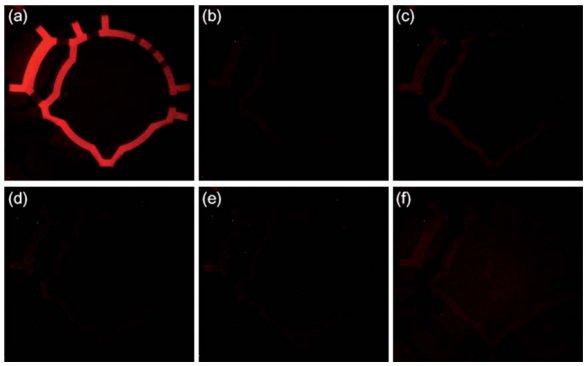


**Fig. 4** Fluorescence images from the reaction between the double-stranded DNA probe and the target DNA. The effect of the length of the target DNA was explored. (a) 50 nt target DNA: perfect match with the Cy3-conjugated sequence of the double-stranded DNA probe. A sequence change occurred between the target DNA and the BHQ2-modified DNA. (b) 40 nt target DNA: partial match from the 5′ end of the Cy3-modified DNA. (c) 40 nt target DNA: partial match from the 3′ end of the Cy3-modified DNA. (d) 30 nt target DNA: partial match from the 5′ end of the Cy3-modified DNA. (e) 30 nt target DNA: partial match from the 3′ end of the Cy3-modified DNA. (f) Water without any target DNA.

fluorescence intensity was strong, as shown in Fig. 4a. This means that the quencher probe strand was displaced by the perfect-matched sequence. Thus, the distance between the Cy3-modified probe DNA and the quencher probe increased and FRET was reduced. For comparison, the short-length target DNA and the double-stranded DNA probe were mixed using a peristaltic pump, and the resulting mixture was monitored (Fig. 4b–e). In these cases, no change in fluorescence intensity was observed when compared with the case where no DNA was used (Fig. 4f). This means that DNA displacement and hybridization did not occur between the Cy3 probe and the short target probe. As a result, the distance between the Cy3 probe and the quencher probe did not change, and FRET was maintained. Since this detection method was based on the thermodynamic difference between the quencher probe and the target, the separation between inconsistency and consistency was feasible for resolution theorem proving.

## DNA computing using the sequential reaction chip

To evaluate the potential of using the developed chip and the sequential reaction for solving resolution theorem proving, eight cases were prepared. Fig. 5a shows all the clauses in which the DNA sequences joined the reaction. In other words, this case was logically inconsistent. All of the DNAs were mixed and introduced into the circular loop. After hybridization and ligation, single-stranded DNA R was injected into the inlet of chamber 1 and was mixed. Exonuclease I was prepared using chamber 2, followed by mixing. In addition, a double-stranded DNA probe was used for detection. A strong fluorescence signal was observed, which implied logical inconsistency (nil). Fig. 5b shows the case where no DNA clauses were used. Distilled water and ligase with a buffer were pumped into the circular chamber through inlet channels, respectively. Then the sequential reaction and fluorescence detection were performed. Since there was no target DNA, the structure of the double-stranded DNA probe was maintained, and no fluorescence signal was detected. This reaction was also performed in the absence of ligase. The DNA clauses were not connected by the ligase, which means that the AND operation was not performed. No fluorescence signal was detected after sequential mixing (Fig. 5c). The various target mixtures were prepared as logically consistent cases. To further

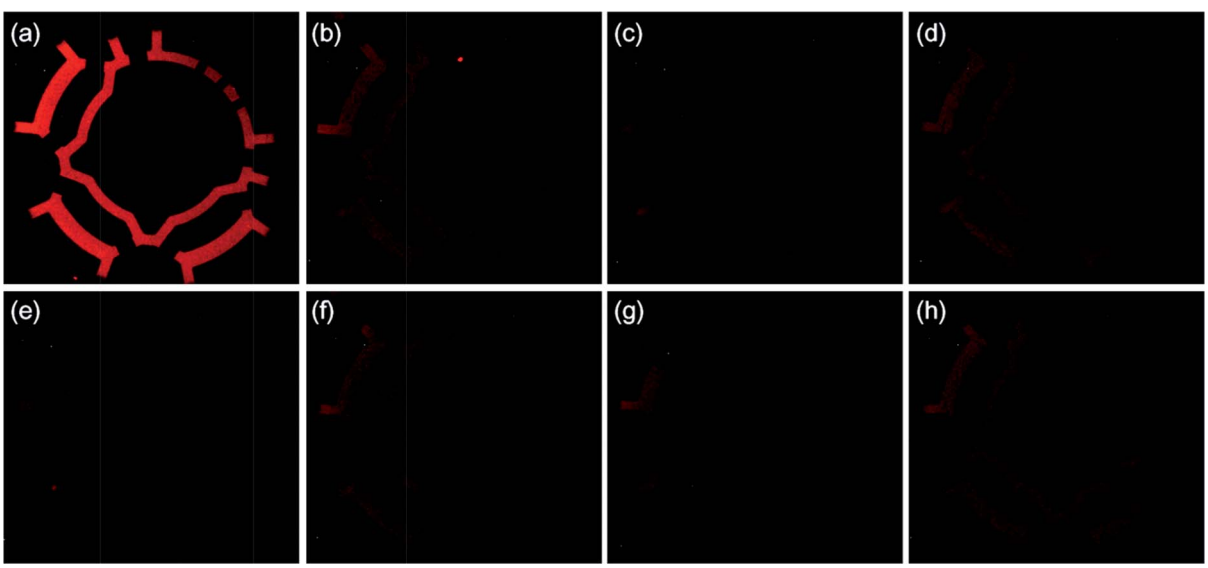


**Fig. 5** Fluorescence micrographs of the sequential reaction chip after the DNA computing process for resolution theorem proving. (a) Logically inconsistent case. All the clauses of the DNA sequence were joined with the reactions. (b) This case did not have any DNA clause as a control experiment; distilled water and ligase were reacted and processed. (c) Ligase was not involved this reaction; since ligase did not exist, the AND operation could not be performed. (d) Only the "$\neg B_1 \vee \neg C$" clause DNA was lost from the whole computing process; thus, this case was logically consistent. After resolution theorem proving, this case could not be driven to nil. (e) "C" was excepted from the reaction. (f) The "$A_1 \vee B_1$" sequence did not participate in the computing reaction. (g) The "$A_0 \vee B_0$" DNA sequence was removed from the reaction. (h) "$\neg B_0 \vee \neg A_1$" did not exist during the whole process.



 

demonstrate this, each case missed a clause DNA, except for "$\neg A_0$". As "$\neg A_0$" was a negation of the conclusion, an additional 10 nt sequence for DNA displacement was attached. Thus, there was no need to test the logical consistency without the "$\neg A_0$" case. In each case, the DNA solution was loaded into the circular loop with ligase and a buffer. For these cases a fluorescence signal was not detected, as shown in Fig. 5d–h. These results indicate that the fluorescent signal generated from the developed microfluidic chip depends on the logical consistency.

## Conclusions

Here, we developed a novel sequential reaction device for the DNA-based logical problem of resolution theorem proving. To solve this problem, a method for encoding logical formulae (clauses) with DNA molecules was used. First, for computation, DNA hybridization and ligation were performed on a chip. Using this method, the number of experiment steps did not vary with the problem size. To monitor the computation results, a sequential reaction was conducted, where the hybridized and ligased DNA was made to undergo: (1) DNA strand displacement *via* toehold reaction; (2) exonuclease I digestion for removing the unreacted DNA and (3) detection using a double-stranded DNA probe.

To separate the consistency of the computation results, the use of intercalating dye was not sufficient. Instead a double-stranded DNA probe was used for sequence-specific detection based on thermodynamics. Efficient mixing between the circular-loop solution and each chamber solution was examined using food dye. In addition, the double-stranded DNA was investigated using the predicted target DNA, and a fluorescence signal was observed when perfect-match target DNA was present.

Finally, computation and detection using sequential reaction were performed for every possible case. As a result, a fluorescence signal was detected only in the logically inconsistent case. Since most DNA computing problems are based on DNA hybridization and ligation, these sequential reactions using a double-stranded DNA probe can be applied to diverse computational problems. Furthermore, the microfluidic system utilizes a solution-based platform. Based on its many advantages, the developed sequential reaction device and method can be applied to many biological and chemical problems, such as ligase detection reaction (LDR), which is a technique that can discriminate a single-base mutation or polymorphisms from a wild-type for medical diagnosis and pathogen detection.[40–42]

## Materials and methods

### Sequence preparation

All single-stranded DNA sequences were purchased from Bioneer (Dajeon, South Korea). The literal and clause oligonucleotides were covalently labeled with phosphate at their 5′ end. The literal "C" oligonucleotide was synthesized with fully phosphorothioate modified backbones. The oligonucleotide pellets were diluted in distilled water to a 500 μM stock concentration and stored at −20 °C until use. The single-stranded DNA R containing toehold for the sequential reaction was synthesized by Bioneer (Table S3, ESI†). For the double-stranded DNA probe, the fluorophore and quencher probes were custom-made by Integrated DNA Technologies, Inc. (Coralville, IA, USA). Stock DNA solutions at a concentration of 100 μM were prepared using distilled water and stored at −20 °C. Before the experiment, hybridization reactions were performed in a 20 μl reaction buffer containing 10 mM $MgCl_2$. The reaction buffer containing a fluorophore and a quencher probe (Cy3 probe: 25 μM; quencher probe: 50 μM) was incubated at 95 °C for 3 min, and the temperature was steadily lowered to 25 °C by 0.5 °C $min^{-1}$, using a thermal cycler (Bio-Rad, Hercules, CA, USA).

### DNA computing on a chip for resolution theorem proving

Before loading the DNA samples into the microfluidic device, equal amounts (100 μM) of single stranded DNA (ssDNA) literals were mixed. Quick T4 DNA ligase (New England Biolabs, MA, USA) and a 10× ligation reaction buffer were mixed. Then each mixed solution was introduced into the microfluidic circular channel for the ligation reaction, respectively. After 30 min, sequential reactions were allowed to proceed in chambers 1, 2, and 3. Each chamber worked independently during the stepwise reaction. In the first step of the sequential reaction, 100 μM single-stranded DNA R, which was used for toehold-mediated DNA displacement, was loaded through the inlet channel of chamber 1 and mixed for 10 min. Then chamber 2 was filled with exonuclease I (New England Biolabs, MA, USA), which degraded ssDNA from 3′ to 5′, for the second reaction, and was merged for 30 min. Finally, the double-stranded DNA probe solution (Cy3 probe: 25 μM; quencher probe: 50 μM) was pumped into chamber 3 and allowed to mix. After 10 min, the fluorescence signal was detected using a fluorescent stereo microscope (SV-6, Zeiss Optics, Jena, Germany), which was equipped with a Peltier-cooled CCD camera to monitor the FRET result.

## Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Ministry of Education, Science and Technology (MEST) (no. 2010K001137, no. 2010-0020821, no. 2011-0000331, no. 2011-0001643).

## Notes and references

1 R. S. Braich, N. Chelyapov, C. Johnson, P. W. K. Rothemund and L. Adleman, *Science*, 2002, **296**, 499–502.
2 D. Faulhammer, A. R. Cukras, R. J. Lipton and L. F. Landweber, *Proc. Natl. Acad. Sci. U. S. A.*, 2000, **97**, 1385–1389.
3 P. W. K. Rothemund, *Nature*, 2006, **440**, 297–302.
4 D. van Noort, P. Wagler and J. S. McCaskill, *Smart Mater. Struct.*, 2002, **11**, 756–760.
5 W. H. Grover and R. A. Mathies, *Lab Chip*, 2005, **5**, 1033–1040.
6 S. Haeberle and R. Zengerle, *Lab Chip*, 2007, **7**, 1094–1110.
7 C. Fang, Y. Wang, N. T. Vu, W.-Y. Lin, Y.-T. Hsieh, L. Rubbi, M. E. Phelps, M. Muschen, Y.-M. Kim, A. F. Chatziioannou, H.-R. Tseng and T. G. Graeber, *Cancer Res.*, 2010, **70**, 8299–8308.
8 J. S. McCaskill, *BioSystems*, 2001, **59**, 125–138.
9 D. van Noort, F.-U. Gast and J. S. McCaskill, *Lect. Notes Comput. Sci. Eng.*, 2002, **2340**, 33–45.
10 D. van Noort and L. F. Landweber, *Lect. Notes Comput. Sci. Eng.*, 2004, **2943**, 190–196.
11 D. van Noort, *Lect. Notes Comput. Sci. Eng.*, 2005, **3384**, 365–374.
12 H. Xie, B. Li, J. Qin, Z. Huang, Y. Zhu and B. Lin, *Electrophoresis*, 2009, **30**, 3514–3518.

13 K.-H. Yea, S. Lee, J. Choo, C.-H. Oh and S. Lee, *Chem. Commun.*, 2006, 1509–1511.
14 J. Jung, L. Chen, S. Lee, S. Kim, G. H. Seong, J. Choo, E. K. Lee, C.-H. Oh and S. Lee, *Anal. Bioanal. Chem.*, 2007, **387**, 2609–2615.
15 A. T.-H. Hsieh, P. J.-H. Pan and A. P. Lee, *Microfluid. Nanofluid.*, 2009, **6**, 391–401.
16 L. Chen, S. Lee, M. Lee, C. Lim, J. Choo, J. Y. Park, S. Lee, S.-W. Joo, K.-H. Lee and Y.-W. Choi, *Biosens. Bioelectron.*, 2008, **23**, 1878–1882.
17 Q. Li, G. Luan, Q. Guo and J. Liang, *Nucleic Acids Res.*, 2002, **30**, e5.
18 R. J. Lipton, *Science*, 1995, **268**, 542–545.
19 S. Kobayashi, *J. Comb. Optim.*, 1999, **3**, 277–299.
20 V. Mihalache, *Proceedings of the IEEE International Conference on Evolutionary Computation*, Indianapolis, 1997.
21 H. Uejima, M. Hagiya and S. Kobayashi, *Lect. Notes Comput. Sci. Eng.*, 2002, **2340**, 308–320.
22 P. Wasiewicz, T. Janczak, J. J. Mulawka and A. Plucienniczak, *Cybern. Syst.*, 2000, **31**, 283–315.
23 G. F. Luger and W. A. Stubblefield, *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*, Benjamin/Cummings, San Francisco, 1993.
24 N. J. Nilsson, *Artificial Intelligence: A New Synthesis*, Morgan Kaufmann, San Fransisco, 1998.
25 I.-H. Lee, K.-A. Yang, J.-H. Lee, J.-Y. Park, Y. G. Chai, J.-H. Lee and B.-T. Zhang, *Nanotechnology*, 2008, **19**, 395103.
26 Z. Galil, *On the Complexity of Resolution Procedures for Theorem Proving*, Cornell University Computer Science Technical Reports, Cornell, 1974.
27 I.-H. Lee, J.-Y. Park, H.-M. Jang, Y.-G. Chai and B.-T. Zhang, *Lect. Notes Comput. Sci. Eng.*, 2003, **2568**, 156–167.
28 S.-Y. Shin, I.-H. Lee, D. Kim and B.-T. Zhang, *IEEE Trans. Evol. Comput.*, 2005, **9**, 143–158.
29 D. Y. Zhang and E. Winfree, *J. Am. Chem. Soc.*, 2009, **131**, 17303–17314.
30 G. Seelig, D. Soloveichik, D. Y. Zhang and E. Winfree, *Science*, 2006, **314**, 1585–1588.
31 B. M. Frezza, S. L. Cockroft and M. R. Ghadiri, *J. Am. Chem. Soc.*, 2007, **129**, 14875–14879.
32 D. Y. Zhang, A. J. Turberfield, B. Yurke and E. Winfree, *Science*, 2007, **318**, 1121–1125.
33 G. Seelig, B. Yurke and E. Winfree, *J. Am. Chem. Soc.*, 2006, **128**, 12211–12220.
34 B. Yurke, A. J. Turberfield, A. P. Mills, Jr, F. C. Simmel and J. L. Neumann, *Nature*, 2000, **406**, 605–608.
35 M. A. Unger, H.-P. Chou, T. Thorsen, A. Scherer and S. R. Quake, *Science*, 2000, **288**, 113–116.
36 S. Kou, H. N. Lee, D. van Noort, K. M. K. Swamy, S. H. Kim, J. H. Soh, K.-M. Lee, S.-W. Nam, J. Yoon and S. Park, *Angew. Chem.*, 2008, **120**, 886–890.
37 S. H. Lee, D. van Noort, J. Y. Lee, B.-T. Zhang and T. H. Park, *Lab Chip*, 2008, **9**, 479–482.
38 H.-W. Rhee, S. H. Lee, I.-S. Shin, S. J. Choi, H. H. Park, K. Han, T. H. Park and J.-I. Hong, *Angew. Chem.*, 2010, **122**, 5039–5043.
39 H.-P. Chou, M. A. Unger and S. R. Quake, *Biomed. Microdevices*, 2001, **3**, 323–330.
40 Z. Peng, S. A. Soper, M. R. Pingle, F. Barany and L. M. Davis, *Anal. Chem.*, 2010, **82**, 9727–9735.
41 R. Favis and F. Barany, *Ann. N. Y. Acad. Sci.*, 2000, **906**, 39–43.
42 M. Khanna, P. Park, M. Zirvi, W. Cao, A. Picon, J. Day, P. Paty and F. Barany, *Oncogene*, 1999, **18**, 27–38.